Sketching meets Krylov in space-time

Valeria Simoncini

Dipartimento di Matematica
Alma Mater Studiorum - Università di Bologna
valeria.simoncini@unibo.it

From joint works with Julian Henning, Davide Palitta, Marcel Schweitzer, Karsten Urban

Large linear systems

Given a PDE and your preferred discretization strategy,

$$\mathcal{A}x = b, \quad \mathcal{A} \in \mathbb{R}^{n\times n}$$

- Krylov subspace methods (CG, MINRES, GMRES, BiCGSTAB, etc.)

$$x \approx x_m = \mathcal{V}_m y_m$$

where $\mathcal{V}_m$ has orthonormal columns spanning $\mathbb{K}_m(\mathcal{A}, b) = \mathrm{span}\{b, \mathcal{A}b, \ldots, \mathcal{A}^{m-1}b\}$

- Preconditioners: find P such that

$$AP^{-1}\widetilde{x} = b \qquad x = P^{-1}\widetilde{x}$$

where AP^{-1} is "easier" to solve with.

Comfort zone

Large linear systems

Given a PDE and your preferred discretization strategy,

$$\mathcal{A}x = b, \quad \mathcal{A} \in \mathbb{R}^{n\times n}$$

- Krylov subspace methods (CG, MINRES, GMRES, BiCGSTAB, etc.)

  $$x \approx x_m = \mathcal{V}_m y_m$$

  where $\mathcal{V}_m$ has orthonormal columns spanning $\mathbb{K}_m(\mathcal{A}, b) = \mathrm{span}\{b, \mathcal{A}b, \ldots, \mathcal{A}^{m-1}b\}$
- Preconditioners: find P such that

  $$AP^{-1}\widetilde{x} = b \qquad x = P^{-1}\widetilde{x}$$

  where AP^{-1} is "easier" to solve with.

Comfort zone

Large linear systems

Given a PDE and your preferred discretization strategy,

$$\mathcal{A}x = b, \quad \mathcal{A} \in \mathbb{R}^{n\times n}$$

- Krylov subspace methods (CG, MINRES, GMRES, BiCGSTAB, etc.)

$$x \approx x_m = \mathcal{V}_m y_m$$

  where $\mathcal{V}_m$ has orthonormal columns spanning $\mathbb{K}_m(\mathcal{A}, b) = \mathrm{span}\{b, \mathcal{A}b, \ldots, \mathcal{A}^{m-1}b\}$

- Preconditioners: find P such that

$$AP^{-1}\widetilde{x} = b \qquad x = P^{-1}\widetilde{x}$$

  where AP^{-1} is “easier” to solve with.

$\boxed{\text{Comfort zone}}$

Heterogeneous variable setting

The differential problem may depend on space variable and

- Time (high quality soln of heat-, wave-type equations, dynamical systems generally)
- Parameters (e.g., coefficients with uncertainty, model tuning)

$$\Downarrow$$

Approximation space in the discretization phase: tensor space

$$\mathcal{H} \times \mathcal{S}$$

with ♣ $\mathcal{H}$: spatial variables
♣ $\mathcal{S}$: time/parameter variables

Algebraic system: $\mathcal{A}$ mixes all components, e.g.,

$$\mathcal{A} = I \otimes A + G^T \otimes I$$

Heterogeneous variable setting

The differential problem may depend on space variable and

- Time (high quality soln of heat-, wave-type equations, dynamical systems generally)
- Parameters (e.g., coefficients with uncertainty, model tuning)

$$\Downarrow$$

Approximation space in the discretization phase: tensor space

$$\mathcal{H} \times \mathcal{S}$$

with ♣ $\mathcal{H}$: spatial variables
♣ $\mathcal{S}$: time/parameter variables

Algebraic system: $\mathcal{A}$ mixes all components, e.g.,

$$\mathcal{A} = I \otimes A + G^T \otimes I$$

Heterogeneous variable setting

The differential problem may depend on space variable and

- Time (high quality soln of heat-, wave-type equations, dynamical systems generally)
- Parameters (e.g., coefficients with uncertainty, model tuning)

$$\Downarrow$$

Approximation space in the discretization phase: tensor space

$$\mathcal{H} \times \mathcal{S}$$

with ♣ $\mathcal{H}$: spatial variables
♣ $\mathcal{S}$: time/parameter variables

Algebraic system: $\mathcal{A}$ mixes all components, e.g.,

$$\mathcal{A} = I \otimes A + G^T \otimes I$$

Identity-preserving algebraic formulations

$$\mathcal{A}x = b, \qquad \mathcal{A} = I \otimes A + G^T \otimes I \qquad \mathcal{A} \in \mathbb{R}^{n\times n}, \text{ with } n = n_A n_G$$

$$\Downarrow$$

$$AX + XG = B, \qquad x = \mathrm{vec}(X), \qquad b = \mathrm{vec}(B), \quad X \in \mathbb{R}^{n_A \times n_G}$$

Pros:

- ✓ Matrices of Smaller dimension ⇒ Reach more complex problems
- ✓ No mixing – Preserve properties of continuous problem
- ✓ Exploit algebraic structure (symmetries, rank properties...)

Identity-preserving algebraic formulations

$$\mathcal{A}x = b, \qquad \mathcal{A} = I \otimes A + G^T \otimes I \qquad \mathcal{A} \in \mathbb{R}^{n\times n}, \text{ with } n = n_A n_G$$

$$\Downarrow$$

$$AX + XG = B, \qquad x = \mathrm{vec}(X), \qquad b = \mathrm{vec}(B), \quad X \in \mathbb{R}^{n_A \times n_G}$$

Pros:

- ✓ Matrices of Smaller dimension $\Rightarrow$ Reach more complex problems
- ✓ No mixing – Preserve properties of continuous problem
- ✓ Exploit algebraic structure (symmetries, rank properties...)

Identity-preserving algebraic formulations

$$\mathcal{A}x = b, \qquad \mathcal{A} = I \otimes A + G^T \otimes I \qquad \mathcal{A} \in \mathbb{R}^{n\times n}, \text{ with } n = n_A n_G$$

$$\Downarrow$$

$$AX + XG = B, \qquad x = \mathrm{vec}(X), \qquad b = \mathrm{vec}(B), \quad X \in \mathbb{R}^{n_A \times n_G}$$

Pros:

- ✓ Matrices of Smaller dimension $\Rightarrow$ Reach more complex problems
- ✓ No mixing – Preserve properties of continuous problem
- ✓ Exploit algebraic structure (symmetries, rank properties...)

Exploiting rank structure

Assume B can be well represented by a low rank matrix.

$$x \in \mathbb{R}^{n_A n_G \times 1} \quad \rightarrow \quad X \approx \widetilde{X} = \begin{bmatrix} X_1 \\ \\ \end{bmatrix} [\ X_2^T\ ]$$

with $X_1 \in \mathbb{R}^{n_A \times k}, X_2 \in \mathbb{R}^{n_G \times k}$ tall, $k \ll n_a, n_G$

Uncover low rank approximate representation!

- Save memory allocations while approximating!
- Different interpretation: approximate soln snapshots (MOR style)
- Recognize roles at the algebraic level: use different approximations for X_1, X_2

Exploiting rank structure

Assume B can be well represented by a low rank matrix.

$$x \in \mathbb{R}^{n_A n_G \times 1} \quad \rightarrow \quad X \approx \widetilde{X} = \begin{bmatrix} X_1 \end{bmatrix} [\ X_2^T\ ]$$

with $X_1 \in \mathbb{R}^{n_A \times k}, X_2 \in \mathbb{R}^{n_G \times k}$ tall, $k \ll n_a, n_G$

Uncover low rank approximate representation!

- Save memory allocations while approximating!
- Different interpretation: approximate soln snapshots (MOR style)
- Recognize roles at the algebraic level: use different approximations for X_1, X_2

Numerical solution of the Sylvester equation

$$AX + XG^T = B$$

Various settings:

- Tiny A and G: Kron will do!
- Small A and G: Bartels-Stewart algorithm (Computes the Schur form of A and G)
- Large A and G: Iterative solution (B low rank)
  - Projection methods
  - ADI (Alternating Direction Iteration)
  - Data sparse approaches (structure-dependent)

Numerical solution of the Sylvester equation

$$AX + XG^T = B$$

Various settings:

- Tiny A and G: Kron will do!
- Small A and G: Bartels-Stewart algorithm (Computes the Schur form of A and G)
- Large A and G: Iterative solution (B low rank)
  - Projection methods
  - ADI (Alternating Direction Iteration)
  - Data sparse approaches (structure-dependent)

Numerical solution of the Sylvester equation

$$AX + XG^T = B$$

Various settings:

- Tiny A and G: Kron will do!
- Small A and G: Bartels-Stewart algorithm (Computes the Schur form of A and G)
- Large A and G: Iterative solution (B low rank)
  - Projection methods
  - ADI (Alternating Direction Iteration)
  - Data sparse approaches (structure-dependent)

Projection-type methods

Assume $B = B_1 B_2^T$.
Given two low dimensional approx spaces $\mathcal{K}_A$, $\mathcal{K}_G$, and V_m, W_m their orthonormal bases let $X_m := V_m Y_m W_m^T$, $X_m \approx X$

Galerkin condition: $R := AX_m + X_m G^T - B_1 B_2^T \quad \perp \quad \mathcal{K}_A \otimes \mathcal{K}_G$

$$V_m^\top R W_m = 0$$

Note: $\mathcal{K}_A$, $\mathcal{K}_G$ tiny wrto $\mathbb{K}(\mathcal{A}, b)$

Projected Sylvester equation:

$$V_m^\top (A V_m Y_m W_m^\top + V_m Y_m W_m^\top G^\top \quad - \quad B_1 B_2^\top) W_m = 0$$
$$(V_m^\top A V_m) Y_m + Y_m (V_m^\top G^\top V_m) \quad - \quad V_m^\top B_1 B_2^\top W_m = 0$$

Early contributions: Saad '90, Jaimoukha & Kasenally '94, for
$\mathrm{range}(V_m) = \mathcal{K}_A = \mathrm{Range}([B_1, AB_1, \ldots, A^{m-1}B_1])$

Projection-type methods

Assume $B = B_1 B_2^T$.
Given two low dimensional approx spaces $\mathcal{K}_A$, $\mathcal{K}_G$, and V_m, W_m their orthonormal bases
let $X_m := V_m Y_m W_m^T$, $X_m \approx X$

Galerkin condition: $\quad R := AX_m + X_m G^T - B_1 B_2^T \quad \perp \quad \mathcal{K}_A \otimes \mathcal{K}_G$

$$V_m^\top R W_m = 0$$

Note: $\mathcal{K}_A$, $\mathcal{K}_G$ tiny wrto $\mathbb{K}(\mathcal{A}, b)$

Projected Sylvester equation:

$$\begin{aligned} V_m^\top (AV_m Y_m W_m^\top + V_m Y_m W_m^\top G^\top \quad &- \quad B_1 B_2^\top) W_m = 0 \\ (V_m^\top A V_m) Y_m + Y_m (V_m^\top G^\top V_m) \quad &- \quad V_m^\top B_1 B_2^\top W_m = 0 \end{aligned}$$

Early contributions: Saad '90, Jaimoukha & Kasenally '94, for
$\mathrm{range}(V_m) = \mathcal{K}_A = \mathrm{Range}([B_1, AB_1, \ldots, A^{m-1} B_1])$

More recent options as approximation space

Enrich space to decrease space dimension

- Extended Krylov subspace

  (Druskin & Knizhnerman '98, Simoncini '07)

- Rational Krylov subspace

$$\mathcal{K}_A = \mathbb{K}_A := \mathrm{Range}([B_1, (A - s_1 I)^{-1} B_1, \ldots, \prod_{j=1}^{m-1} (A - s_j I)^{-1} B_1])$$

  usually, $\{s_1, \ldots, s_{m-1}\} \subset \mathbb{C}^+$ chosen either a-priori or dynamically
  (form matrix equations, Druskin & Simoncini '11)

In both cases, for Range(V_m) $= \mathcal{K}_A$, Range(W_m) $= \mathcal{K}_{G^\top}$ projected Lyapunov equation:

$$(V_m^\top A V_m) Y_m + Y_m (W_m^\top G^\top W_m) - V_m^\top B_1 B_2^\top W_m = 0 \qquad X_m = V_m Y_m W_m^\top$$

More recent options as approximation space

Enrich space to decrease space dimension

- Extended Krylov subspace

  (Druskin & Knizhnerman '98, Simoncini '07)

- Rational Krylov subspace

$$\mathcal{K}_A = \mathbb{K}_A := \mathrm{Range}([B_1, (A - s_1 I)^{-1} B_1, \ldots, \prod_{j=1}^{m-1} (A - s_j I)^{-1} B_1])$$

usually, $\{s_1, \ldots, s_{m-1}\} \subset \mathbb{C}^+$ chosen either a-priori or dynamically
(form matrix equations, Druskin & Simoncini '11)

In both cases, for $\mathrm{Range}(V_m) = \mathcal{K}_A$, $\mathrm{Range}(W_m) = \mathcal{K}_{G^\top}$ projected Lyapunov equation:

$$(V_m^\top A V_m) Y_m + Y_m (W_m^\top G^\top W_m) - V_m^\top B_1 B_2^\top W_m = 0 \qquad X_m = V_m Y_m W_m^\top$$

More recent options as approximation space

Enrich space to decrease space dimension

- Extended Krylov subspace

  (Druskin & Knizhnerman '98, Simoncini '07)

- Rational Krylov subspace

$$\mathcal{K}_A = \mathbb{K}_A := \mathrm{Range}([B_1, (A - s_1 I)^{-1} B_1, \ldots, \prod_{j=1}^{m-1} (A - s_j I)^{-1} B_1])$$

  usually, $\{s_1, \ldots, s_{m-1}\} \subset \mathbb{C}^+$ chosen either a-priori or dynamically
  (form matrix equations, Druskin & Simoncini '11)

In both cases, for Range$(V_m) = \mathcal{K}_A$, Range$(W_m) = \mathcal{K}_{G^\top}$ projected Lyapunov equation:

$$(V_m^\top A V_m) Y_m + Y_m (W_m^\top G^\top W_m) - V_m^\top B_1 B_2^\top W_m = 0 \qquad X_m = V_m Y_m W_m^\top$$

A first example. Space-time discretization

$$u_t = \mathcal{L}(u), \qquad \mathcal{L}(u) = -\Delta u + 10xu_x + 10yu_y, \quad (x,y,z) \in (0,1)^3 \quad u(*,t_0) = 0, f = 1$$

Crank-Nicolson type discretization in time, Finite Differences in space

$\Rightarrow$ One-sided approximation, only in space (no dim reduction in time)

CPU time

n_A	n_G	Crank-Nic	RKSM (dim)
15625	400	5.1	1.1 (20)
125000	400	75.7	18.5 (25)
287496	400	314.5	63.8 (28)

$$AX + XG^T = B_1 B_2^T, \qquad X \approx X_1 X_2^T = V_m Y = V_m Y_1 Y_2^T$$

Pros and Cons of Rational Krylov spaces

Pros

- Captures minimal rank on the fly
- Generally very efficient on large sparse problems
- Parameter-free in practice

Cons

- Increasingly more expensive on denser problems (3D)
- Orthogonalization of long vectors

Pros and Cons of Rational Krylov spaces

Pros

- Captures minimal rank on the fly
- Generally very efficient on large sparse problems
- Parameter-free in practice

Cons

- Increasingly more expensive on denser problems (3D)
- Orthogonalization of long vectors

Sketching strategies. Subspace embedding.

A formidable, probability-based, data reduction strategy, applicable to a large variety of settings

A $(1 \pm \varepsilon)$ ℓ_2-subspace embedding for the tall matrix $V \in \mathbb{R}^{n \times k}$ is a matrix S such that, for all $x \in \mathbb{R}^k$,

$$(1-\varepsilon)\|Vx\|_2^2 \leq \|SVx\|_2^2 \leq (1+\varepsilon)\|Vx\|_2^2$$

To build a "feasible" S :

- S needs to have small number of rows, r
- The products SV should be cheap
- Probabilistic confidence on the quality of S

See, e.g., David Woodruff (2014), Martinsson and Tropp, Acta Num. (2020)

Sketching strategies. Subspace embedding.

A formidable, probability-based, data reduction strategy, applicable to a large variety of settings

A $(1 \pm \varepsilon)$ ℓ_2-subspace embedding for the tall matrix $V \in \mathbb{R}^{n \times k}$ is a matrix S such that, for all $x \in \mathbb{R}^k$,

$$(1-\varepsilon)\|Vx\|_2^2 \le \|SVx\|_2^2 \le (1+\varepsilon)\|Vx\|_2^2$$

To build a "feasible" S :

- S needs to have small number of rows, r
- The products SV should be cheap
- Probabilistic confidence on the quality of S

See, e.g., David Woodruff (2014), Martinsson and Tropp, Acta Num. (2020)

Sketching strategies. Subspace embedding.

A formidable, probability-based, data reduction strategy, applicable to a large variety of settings

A $(1 \pm \varepsilon)$ ℓ_2-subspace embedding for the tall matrix $V \in \mathbb{R}^{n \times k}$ is a matrix S such that, for all $x \in \mathbb{R}^k$,

$$(1-\varepsilon)\|Vx\|_2^2 \leq \|SVx\|_2^2 \leq (1+\varepsilon)\|Vx\|_2^2$$

To build a "feasible" S :

- S needs to have small number of rows, r
- The products SV should be cheap
- Probabilistic confidence on the quality of S

See, e.g., David Woodruff (2014), Martinsson and Tropp, Acta Num. (2020)

The Subsampled Randomized Hadamard Transform

- $\Rightarrow$ If S is an $r \times n$ matrix of i.i.d. $N(0, 1/r)$ with $r = \mathcal{O}(k/\varepsilon^2)$ then S is $(1 \pm \varepsilon)$ embedding
- $\Rightarrow$ If S taken from fast Johnson-Lindenstrauss transforms, then SV only costs $\mathcal{O}(nk)$ (Tamás Sarlós)

A convenient choice giving a fast Johnson-Lindenstrauss transform:

$$Sv := \frac{1}{\sqrt{rn}} PCDv, \qquad S \text{ is an } r \times n \text{ matrix}$$

(Randemacher operator)
with
D "rotation" (diagonal matrix from uniform distr. in $(-1, 1)$)
C fast cosine transform
P coordinate sampling

See, e.g., David Woodruff (2014), Martinsson and Tropp, Acta Num. (2020)

The Subsampled Randomized Hadamard Transform

$\Rightarrow$ If S is an $r \times n$ matrix of i.i.d. $N(0, 1/r)$ with $r = \mathcal{O}(k/\varepsilon^2)$ then S is $(1 \pm \varepsilon)$ embedding

$\Rightarrow$ If S taken from fast Johnson-Lindenstrauss transforms, then SV only costs $\mathcal{O}(nk)$ (Tamás Sarlós)

A convenient choice giving a fast Johnson-Lindenstrauss transform:

$$Sv := \frac{1}{\sqrt{rn}} PCDv, \qquad S \text{ is an } r \times n \text{ matrix}$$

(Randemacher operator)
with
D "rotation" (diagonal matrix from uniform distr. in $(-1, 1)$)
C fast cosine transform
P coordinate sampling

See, e.g., David Woodruff (2014), Martinsson and Tropp, Acta Num. (2020)

Sketching meets Krylov

How can we use subspace embedding in our context to get $X_m = V_m Y_m W_m^T$?

- Compute a "cheap" space $\mathit{range}(V_m)$
- Avoid orth and storing V_m
- $V_m \quad \Rightarrow \quad SV_m$
- Orthogonalize shorter vectors in SV_m

(similarly for W_m)

Technical details:

♣ Local orthogonality in V_m, W_m ("truncated basis")

♣ Two-pass strategy to recover $X_m = (V_m Y_m) W_m^T$

Currently NLA 'hot topic'. In the "Krylov world", Balabanov, Cortinovis, Grigori, Guettel, Kressner, Nakatsukasa, Nouy, Schweitzer, Timsit, Tropp, etc.

Sketching meets Krylov

How can we use subspace embedding in our context to get $X_m = V_m Y_m W_m^T$?

- Compute a "cheap" space *range*(V_m)
- Avoid orth and storing V_m
- $V_m \quad \Rightarrow \quad SV_m$
- Orthogonalize shorter vectors in SV_m

(similarly for W_m)

Technical details:

♣ Local orthogonality in V_m, W_m ("truncated basis")

♣ Two-pass strategy to recover $X_m = (V_m Y_m) W_m^T$

Currently NLA 'hot topic'. In the "Krylov world", Balabanov, Cortinovis, Grigori, Guettel, Kressner, Nakatsukasa, Nouy, Schweitzer, Timsit, Tropp, etc.

The sketched problem

Instead of imposing Galerkin condition $V_m^T R_m W_m = 0$ and solve

$$(V_m^\top A V_m) Y_m + Y_m (W_m^\top G^\top W_m) = V_m^\top B_1 B_2^\top W_m \qquad X_m = V_m Y_m W_m^\top$$

We impose the "sketched" Galerkin condition $Q_m^T S_V R_m S_W P_m = 0$ and solve

$$(Q_m^\top S A V_m) Y_m T_W^T + T_V Y_m (W_m^\top G^\top P_m) = Q_m^\top S_V B_1 B_2^\top S_W^T P_m$$

where $S_V V_m = Q_m T_V$, $S_W W_m = P_m T_W$ are QR factorizations

Work in progress with Davide Palitta, Marcel Schweitzer

The same problem seen earlier

$u_t - \mathcal{L}(u) = f, \qquad \mathcal{L}(u) = -\Delta u + 10xu_x + 10yu_y, \quad (x, y, z) \in (0,1)^3$
$u(*, t_0) = 0, f = 1$

Crank-Nicolson type discretization in time, Finite differences in space

$\Rightarrow$ One-sided approximation, only in space (no dim reduction in time)

Truncation: 5, Sketched space dim: 1000 $(= 2m_{\max})$

CPU time

n_A	n_G	Crank-Nic	RKSM (dim)	Sketched Krylov (dim)
15625	400	5.08	1.15 (20)	1.04 (180)
125000	400	75.71	18.50 (25)	10.72 (260)
287496	400	314.54	63.82 (28)	39.53 (360)

$$AX + XG^T = B_1 B_2^T, \qquad X \approx X_1 X_2^T$$

All-at-once heat equation

$$u_t + \Delta u = f \qquad u(0) = 0$$

Variational formulation

$$\text{find } u \in U : \quad b(u,v) = \langle f, v\rangle \quad \text{for all } v \in V$$

where $U := H^1_{(0)}(\mathcal{I}; X') \cap L_2(\mathcal{I}, X),\ X := H^1_0(\Omega),\ V := L_2(\mathcal{I}; X)$

$b(u,v) := \int_0^{\mathcal{T}} \int_\Omega u_t(t,x)\, v(t,x)\, dx\, dt + \int_0^{\mathcal{T}} a(u(t), v(t))\, dt$

$\langle f, v\rangle := \int_0^{\mathcal{T}} \int_\Omega f(t,x)\, v(t,x)\, dx\, dt.$

Discretization: Petrov-Galerkin method with trial and test spaces $U_\delta \subset U$, $V_\delta \subset V$

$$\text{find } u_\delta \in U_\delta : \quad b(u_\delta, v_\delta) = \langle f, v_\delta\rangle \quad \text{for all } v_\delta \in V_\delta$$

with $U_\delta := S_{\Delta t} \otimes X_h$, $V_\delta = Q_{\Delta t} \otimes X_h$ where

$S_{\Delta t}$: piecewise linear FE on $\mathcal{I}$
$Q_{\Delta t}$: piecewise constant FE on $\mathcal{I}$
X_h : any conformal space, e.g., p.w. linear FE

♣ Well-posedness (discrete inf-sup cond) depends on the choice of U_δ, V_δ

Remark: This discretization coincides with Crank–Nicolson scheme if trapezoidal approximation of the rhs temporal integration is used

All-at-once heat equation

$$u_t + \Delta u = f \qquad u(0) = 0$$

Variational formulation

$$\text{find } u \in U : \quad b(u,v) = \langle f, v\rangle \quad \text{for all } v \in V$$

where $U := H^1_{(0)}(\mathcal{I}; X') \cap L_2(\mathcal{I}, X),\ X := H^1_0(\Omega),\ V := L_2(\mathcal{I}; X)$
$b(u,v) := \int_0^{\tau} \int_\Omega u_t(t,x)\, v(t,x)\, dx\, dt + \int_0^{\tau} a(u(t), v(t))\, dt$
$\langle f, v\rangle := \int_0^{\tau} \int_\Omega f(t,x)\, v(t,x)\, dx\, dt.$

Discretization: Petrov-Galerkin method with trial and test spaces $U_\delta \subset U$, $V_\delta \subset V$

$$\text{find } u_\delta \in U_\delta : \quad b(u_\delta, v_\delta) = \langle f, v_\delta\rangle \quad \text{for all } v_\delta \in V_\delta$$

with $U_\delta := S_{\Delta t} \otimes X_h$, $V_\delta = Q_{\Delta t} \otimes X_h$ where

$S_{\Delta t}$: piecewise linear FE on $\mathcal{I}$
$Q_{\Delta t}$: piecewise constant FE on $\mathcal{I}$
X_h : any conformal space, e.g., p.w. linear FE

♣ Well-posedness (discrete inf-sup cond) depends on the choice of U_δ, V_δ

Remark: This discretization coincides with Crank–Nicolson scheme if trapezoidal approximation of the rhs temporal integration is used

The final linear system

$$B_\delta^\top u_\delta = f_\delta$$

where

$$[B_\delta]_{(k,i),(\ell,j)} = (\dot{\sigma}^k, \tau^\ell)_{L_2(\mathcal{I})}\,(\phi_i,\phi_j)_{L_2(\Omega)} + (\sigma^k,\tau^\ell)_{L_2(\mathcal{I})}\,a(\phi_i,\phi_j),$$
$$[f_\delta]_{(\ell,j)} = (f, \tau^\ell \otimes \phi_j)_{L_2(\mathcal{I};H)}$$

that is, $B_\delta = D_{\Delta t} \otimes M_h + C_{\Delta t} \otimes K_h$

Remark: We approximate f_δ to achieve full tensor-product structure

Resulting generalized Sylvester equation:

$$K_h \mathbf{U}_\delta C_{\Delta t} + M_h \mathbf{U}_\delta D_{\Delta t} = F_\delta, \qquad \text{with} \quad F_\delta = [g_1,\ldots,g_P][h_1,\ldots,h_P]^\top$$

$$\boxed{F_\delta \text{ matrix of low rank} \quad \Rightarrow \quad \mathbf{U}_\delta \text{ approx by low rank matrix } \widetilde{\mathbf{U}}_\delta}$$

(Julian Henning, Davide Palitta, V. S., Karsten Urban, 2021)

Sketching strategies. Preliminary runs. 1

$$M_h^{-1} K_h \mathbf{U}_\delta + \mathbf{U}_\delta D_{\Delta t} C_{\Delta t}^{-1} = M_h^{-1} F_\delta C_{\Delta t}^{-1}$$

($M_h^{-1} K_h$ not formed explicitly) corresponds to

$$AX + XG^T = B_1 B_2^T, \qquad X \approx X_1 X_2^T$$

We solve by only reducing the space variable (One-sided algebraic approx)

Truncation: 5, Sketched space dim: 1000 ($= 2m_{\max}$)

CPU time (secs)

n_A	n_G	Crank-Nic	RKSM (dim)	Sketched Krylov (dim)
9472	1000	8.4	4.3 (17)	3.1 (180)
13085	1000	10.8	6.40 (17)	4.7 (200)
19126	1000	14.5	11.7 (18)	6.3 (220)
29430	1000	20.8	19.4 (17)	10.1 (240)
46545	1000	35.1	52.9 (20)	21.5 (320)
82270	1000	53.3	149.9 (21)	42.2 (360)
163195	1000	165.6	356.2 (20)	104.5 (460)
393968	1000	504.9	1634.2 (21)	388.5 (620)

(approximate solution rank: 17-20)

Sketching strategies. Preliminary runs. 2

Truncation: 5, Sketched space dim: 1000 ($= 2m_{\max}$)

CPU time (secs), mass matrix lumping in space

n_A	n_G	Crank-Nic	RKSM (dim)	Sketched Krylov (dim)
9472	1000	8.3	1.5 (15)	0.9 (120)
13085	1000	14.1	1.7 (15)	1.6 (120)
19126	1000	18.9	2.2 (16)	1.6 (140)
29430	1000	29.1	2.7 (16)	2.1 (160)
46545	1000	43.9	4.1 (16)	3.4 (180)
82270	1000	83.2	8.5 (19)	7.1 (220)
163195	1000	225.2	18.9 (19)	17.8 (260)
393968	1000	807.8	53.8 (16)	60.2 (360)

(approximate solution rank: 17-20)

Conclusions

Sketching:

- Very promising strategy for big data
- Easily applicable in matrix computations contexts
- Analysis of theoretical impacts is ongoing in various communities

- In our context: pushes dimension limits ahead

Visit: www.dm.unibo.it/~simoncin
Email address: valeria.simoncini@unibo.it

Conclusions

Sketching:

- Very promising strategy for big data
- Easily applicable in matrix computations contexts
- Analysis of theoretical impacts is ongoing in various communities

- In our context: pushes dimension limits ahead

Visit: www.dm.unibo.it/~simoncin
Email address: valeria.simoncini@unibo.it

Conclusions

Sketching:

- Very promising strategy for big data
- Easily applicable in matrix computations contexts
- Analysis of theoretical impacts is ongoing in various communities

- In our context: pushes dimension limits ahead

Visit: `www.dm.unibo.it/~simoncin`
Email address: `valeria.simoncini@unibo.it`